

Nattan MAISONNEUVE–FLEURY
BTS SIO 1
Groupe B

16/09/2025

Compte Rendu - Java

Sommaire :

Introduction :	1
Exercice 4 - algorithme de moyenne dans un tableau	1
Question 3 - je suis mineur/majeur/vétérane ?	3
TP - Java	5
QCM -	5
Exercice de programmation :	6
debug 1	8
debug 2 :	
public class Debug2	8
Debug 3 :	9
Debug 4 :	10
Cas pratique :	10

Introduction :

Aujourd'hui nous allons apprendre de nouvelles balises en java mais surtout voir comment réparer des erreurs.

Exercice 4 - algorithme de moyenne dans un tableau

Le but ici est de créer un code où l'utilisateur va entrer 4 notes. Celles-ci seront ensuite calculées sous forme de moyenne.

Voici le code ainsi que le résultat :

```
1 package test;
2
3 import java.util.Scanner;
4
5 public class code123456 {
6     public static void main(String[] args) {
7
8         double[] tabl = new double[4];
9         double moy;
10
11         Scanner scanner = new Scanner(System.in);
12
13         System.out.print("Entrez la note 1 : ");
14         tabl[0] = scanner.nextDouble();
15
16         System.out.print("Entrez la note 2 : ");
17         tabl[1] = scanner.nextDouble();
18
19         System.out.print("Entrez la note 3 : ");
20         tabl[2] = scanner.nextDouble();
21
22         System.out.print("Entrez la note 4 : ");
23         tabl[3] = scanner.nextDouble();
24
25         moy = (tabl[0] + tabl[1] + tabl[2] + tabl[3]) / 4;
26
27         System.out.println("Votre moyenne est de : " + moy);
28
29         scanner.close();
30     }
31 }
32
```

Entrez la note 1 : 10
Entrez la note 2 : 15
Entrez la note 3 : 15
Entrez la note 4 : 20
Votre moyenne est de : 15.0

Dans ce code, nous voyons plusieurs ligne :

1. Cette ligne nous dit que l'on se trouve dans le dossier test :
`package test;`
2. permet d'importer la classe **Scanner** du package **java.util**. Cette classe est utilisée pour faciliter la lecture des entrées utilisateur à partir de la console :
`import java.util.Scanner;`
3. celle ci nous dis de de prendre les informations dans le fichier code123456.java :
`public class code123456 {`
4. Ici, on définit la méthode principale d'un programme Java. Elle est le point d'entrée où l'exécution du programme commence :
`public static void main(String[] args) {`
5. On déclare et initialise un tableau de 4 éléments de type **double** :
`double[] tabl = new double[4];`
6. Puis, on déclare une variable nommée **moy** de type **double** :
`double moy;`
7. On crée un objet **Scanner** nommé **scanner** pour lire des entrées depuis la console:
`Scanner scanner = new Scanner(System.in);`
8. Ces lignes permettent à l'utilisateur de saisir quatre notes, une par une, qui seront ensuite stockées dans le tableau **tabl** :
`System.out.print("Entrez la note 1 : ");
tabl[0] = scanner.nextDouble();

System.out.print("Entrez la note 2 : ");
tabl[1] = scanner.nextDouble();

System.out.print("Entrez la note 3 : ");
tabl[2] = scanner.nextDouble();

System.out.print("Entrez la note 4 : ");
tabl[3] = scanner.nextDouble();`
9. Maintenant il faut calculer la moyenne des quatre notes saisies et l'assigner à la variable **moy** :
`moy = (tabl[0] + tabl[1] + tabl[2] + tabl[3]) / 4;`
10. Il ne nous reste plus qu'à afficher la moyenne calculée à l'utilisateur :
`System.out.println("Votre moyenne est de : " + moy);`

11. Enfin, fermer l'objet **scanner** après avoir terminé l'utilisation :

```
scanner.close();|
```

Question 3 - je suis mineur/majeur/vétérant ?

Le but ici est de créer un code où l'utilisateur va rentrer son âge. Grâce à cela on va pouvoir savoir si l'utilisateur est mineur, majeur, vétérant.

Voici le code :

```
1 package test;
2
3 import java.util.Scanner;
4
5 public class code123456 {
6     public static void main(String[] args) {
7         int age;
8
9         Scanner scanner = new Scanner(System.in);
10
11         System.out.print("Veuillez saisir votre âge : ");
12         age = scanner.nextInt();
13
14         if (age > 39) {
15             System.out.println("Vétérant");
16         } else if (age > 17) {
17             System.out.println("Majeur");
18         } else {
19             System.out.println("Mineur");
20         }
21
22         scanner.close();
23     }
24 }
25 |
```

```
Veuillez saisir votre âge : 45
Vétérant
```

Dans ce code, nous voyons plusieurs ligne :

1. Cette ligne nous dit que l'on se trouve dans le dossier test :

```
package test;
```
2. permet d'importer la classe **Scanner** du package **java.util**. Cette classe est utilisée pour faciliter la lecture des entrées utilisateur à partir de la console :

```
import java.util.Scanner;
```

3. celle ci nous dis de de prendre les informations dans le fichier code123456.java :

```
public class code123456 {
```

4. Ici, on définit la méthode principale d'un programme Java. Elle est le point d'entrée où l'exécution du programme commence :

```
public static void main(String[] args) {
```

5. On déclare une variable nommée **age** de type **int** :

```
int age;
```

6. On crée un objet **Scanner** nommé **scanner** pour lire des entrées depuis la console:

```
Scanner scanner = new Scanner(System.in);
```

7. Puis on affiche un message dans la console pour demander à l'utilisateur de saisir son âge :

```
System.out.print("Veuillez saisir votre âge : ");
```

8. Cette ligne permet de lire un entier (un nombre entier) saisi par l'utilisateur et de l'assigner à la variable **âge** :

```
age = scanner.nextInt();
```

9. Ce bloc de code utilise une structure **if-else if-else** pour afficher un message en fonction de l'âge de la personne.

```
if (age > 39) {  
    System.out.println("Vétéran");  
} else if (age > 17) {  
    System.out.println("Majeur");  
} else {  
    System.out.println("Mineur");  
}
```

Voici une explication plus détaillée :

if (age > 39) : Si l'âge est supérieur à 39, le programme affiche "**Vétéran**".

else if (age > 17) : Si l'âge n'est pas supérieur à 39 mais est supérieur à 17, le programme affiche "**Majeur**". (Cela s'applique donc aux personnes adultes mais de moins de 40 ans).

else : Si aucune des conditions précédentes n'est remplie (c'est-à-dire si l'âge est 17 ou moins), le programme affiche "**Mineur**".

10. Enfin, fermer l'objet **scanner** après avoir terminé l'utilisation :

```
scanner.close();
```

TP - Java

QCM -

- 1) **Le langage machine le plus basique niveau circuit est :**
Le langage machine
- 2) **Les langages qui permettent d'utiliser un vocabulaire qui utilise les termes : read, write ou add sont :**
Haut niveau
- 3) **Les règles du langage de programmation constituent :**
la syntaxe
- 4) **Un _____ traduit les instructions de langage de haut niveau en code machine**
un compilateur
- 5) **Les emplacements de mémoire nommés de l'ordinateur sont appelés :**
adresses
- 6) **Les opérations individuelles utilisées dans un programme informatique sont souvent regroupées en unités logiques appelées :**
procédures
- 7) **Une instance de classe est :**
un objet
- 8) **Java a une architecture :**
neutre
- 9) **Vous devez compiler les classes écrites en Java dans :**
un bytecode
- 10) **Toutes les instructions de programmation Java doivent se terminer par :**
un point-virgule

Exercice de programmation :

- 1) **Chacun des identifiants de classe suivant est-il légal, légal et conventionnel ou illégal :**
 - a. **maClasse** - légal et conventionnel
 - b. **void** - illégal (mot interdit car utiliser en java comme commande)
 - c. **Golden Retriever** - illégal (a un espace)
 - d. **invoice#** - illégal (a un hashtag)
 - e. **36535CodePostal** - légal et conventionnel
 - f. **Appartement** - légal et conventionnel
 - g. **Fruit** - légal et conventionnel
 - h. **8888** - illégal (commence par un chiffre)
 - i. **EcranTotal()** - illégal (a des apostrophe)
 - j. **Acompte_recevable** - légal et conventionnel

- 2) **Chacun des identifiants de méthode suivant est-il légal, légal et conventionnel ou illégal?**
 - a. **associationRoles()** - légal et conventionnel
 - b. **void()** - illégal (mot interdit car utiliser en java comme commande)
 - c. **Golden Retriever()** - illégal (a un espace)
 - d. **invoice#()** - illégal (a un hashtag)
 - e. **24500CodePostal()** - illégal (commence par un chiffre)
 - f. **PayrollApp()** - légal et conventionnel
 - g. **getReady()** - légal et conventionnel
 - h. **911()** - illégal (commence par un chiffre)
 - i. **EcranTotal()** - légal et conventionnel
 - j. **Acompte_Recevable()** - légal et conventionnel

- 3) **Écrivez, compilez et testez une classe qui utilise une instruction println() pour afficher quatre lignes des paroles de votre chanson préférée. Enregistrer la classe sous ParoleChanson.java**

```
package test;

public class ParoleChanson {
    public static void main(String[] args) {
        System.out.println("A partir de maintenant");
        System.out.println("Plus rien n'a vraiment d'importance");
        System.out.println("Plus d'importance");
        System.out.println("A partir de maintenant");
    }
}
```

A partir de maintenant
Plus rien n'a vraiment d'importance
Plus d'importance
A partir de maintenant

la commande println veut dire : écrire.....

- 4) Écrivez, compilez et testez une classe qui affiche le modèle illustré :

```
package test;

public class TablesEtChaises {
    public static void main(String[] args) {
        System.out.println("x                x");
        System.out.println("x                x");
        System.out.println("x      xxxxxxxxxxxx      x");
        System.out.println("xxxxx x                x xxxxx");
        System.out.println("x  x x                x x  x");
        System.out.println("x  x x                x x  x");
    }
}
```

```

      X                      X
      X                      X
      X      xxxxxxxxxxxx      X
      xxxxx x                x xxxxx
      x  x x                x x  x
      x  x x                x x  x

```

- 5) Écrivez, compilez et testez une classe qui affiche le modèle illustré :

```
package test;

public class triangle {
    public static void main(String[] args) {
        System.out.println("      T      ");
        System.out.println("     TTT     ");
        System.out.println("    TTTT    ");
        System.out.println("   TTTTTT   ");
        System.out.println("  TTTTTTTT  ");
        System.out.println(" TTTTTTTTTT ");
        System.out.println("TTTTTTTTTTTT");
    }
}
```

```

      T
     TTT
    TTTT
   TTTTTT
  TTTTTTTT
 TTTTTTTTTT
TTTTTTTTTTTT

```


debug 1

```
| public class Debug1  
| /* This program displays a greeting */  
| public static void main(String[] args)  
| {  
|   System.out.print("Salut).  
| }
```

Ici, nous voyons plusieurs erreur de base, voici le corrigé en rouge

```
| public class debug1 {  
| /* This program displays a greeting */  
| public static void main(String[] args)  
|     {  
|         System.out.println("Salut");  
|     }  
| }
```

debug 2 :

```
| public class Debug2  
| {  
|   /* This program displays some output  
|   public static void main(String args)  
|   {  
|     System.out.println("Programmer en java est fun.");  
|     System.out.println("Faire un programme");  
|     System.out.println("peut être un challenge.");  
|     System.out.prnitln("mais quand la syntaxe est correcte.");  
|     System.out.println("c'est satisfaisant");  
|   }  
| }
```

Ici, nous voyons plusieurs erreur, dont une grosse qui passe l'entièreté du code en commentaire, voici le corrigé en rouge :

```
| public class Debug2
| {
| // This program displays some output
| public static void main(String[] args)
| {
|     System.out.println("Programmer en java est fun.");
|     System.out.println("Faire un programme");
|     System.out.println("peut être un challenge,");
|     System.out.println("mais quand la syntaxe est correcte,");
|     System.out.println("c'est satisfaisant");
| }
| }
```

Debug 3 :

```
| public class Debug3
| {
| public static void main(String[] args)
| {
|     System.Out.println("Derrière la rivière");
|     system.out.println("et au dela du bois");
|     SysTem.Out.println("à la maison du garde nous irons");
| }
| }
```

Ici, nous avons plusieurs erreurs de majuscule, mais aussi un zéro à la place d'un o.
Voici les erreurs corrigés en rouge :

```
| public class Debug3
| {
| public static void main(String[] args)
| {
|     System.out.println("Derrière la rivière");
|     System.out.println("et au dela du bois");
|     System.out.println("à la maison du garde nous irons");
| }
| }
```

Debug 4 :

```
| import javax.swing.JOptionPane;  
| public class Debug4  
| {  
|     public static main(String[] args)  
|     {  
|         JOptionPane.showMessageDialog(null, 1er GUI program!)  
|     }  
| }
```

Ici il y a encore quelque erreurs, voici le corrigé en rouge :

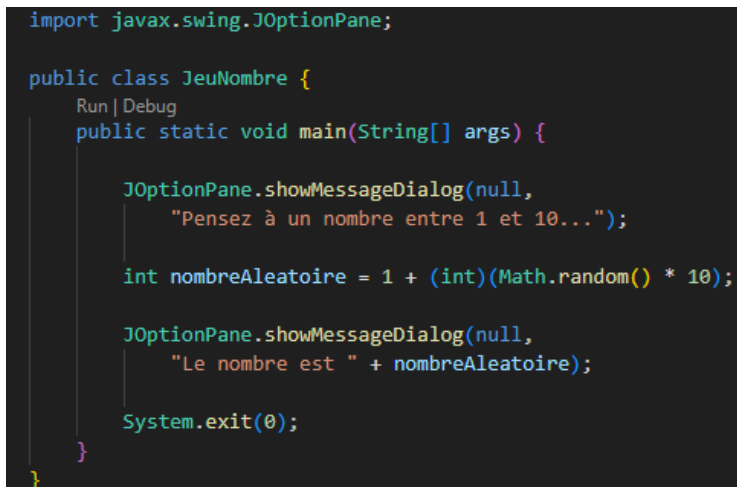
```
| import javax.swing.JOptionPane;  
| public class Debug4  
| {  
|     public static void main(String[] args)  
|     {  
|         JOptionPane.showMessageDialog(null, "1er GUI program");  
|     }  
| }
```

Il faut bien vérifier que le JDK est bien téléchargé depuis la librairie et tout fonctionne.

Cas pratique :

Le but ici est de créer un "jeu" où l'on va penser à un numéro entre 1 et 10. Après, le script va donner un nombre aléatoire et il y aura donc 1 chance sur 10 que se soit le même nombre.

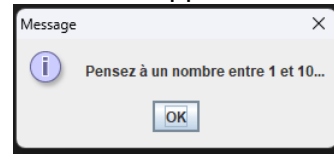
Pour ce faire, nous avons plusieurs lignes de commande à écrire



```
import javax.swing.JOptionPane;  
  
public class JeuNombre {  
    Run | Debug  
    public static void main(String[] args) {  
        JOptionPane.showMessageDialog(null,  
            "Pensez à un nombre entre 1 et 10...");  
  
        int nombreAleatoire = 1 + (int)(Math.random() * 10);  
  
        JOptionPane.showMessageDialog(null,  
            "Le nombre est " + nombreAleatoire);  
  
        System.exit(0);  
    }  
}
```

Je ne vais pas revenir sur les premières lignes car nous les avons déjà vu précédemment cependant nous avons plusieurs nouvelles lignes un peu compliquées.

La première ligne `JOptionPane.showMessageDialog(null....` fais apparaître un message à l'écran en disant "Pensez à un nombre entre 1 et 10.



Ensuite, avec la ligne `int....` le programme va choisir un nombre aléatoire grâce à des formules mathématique aléatoire (`random`) qu'il va stocker dans une variable `nombreAleatoire`.

Enfin, le programme va envoyer un message sur l'écran en donnant le nombre grâce à la variable.

Pour finir, le système se stop.

Cas problèmes :

Dans ce premier problèmes nous devons faire apparaître la devise, voici le code et le résultat :

```
public class Yummy2 {  
    Run | Debug  
    public static void main(String[] args) {  
        System.out.println("Yummy prépare les meilleurs plats pour vos fêtes");  
    }  
}
```

```
Yummy prépare les meilleurs plats pour vos fêtes
```

Dans le deuxième, nous devons afficher la devise entouré d'astérisque

```
public class Yummy2 {  
    Run | Debug  
    public static void main(String[] args) {  
        System.out.println("*****");  
        System.out.println("*Yummy prépare les meilleurs plats pour vos fêtes*");  
        System.out.println("*****");  
    }  
}
```

```
*****  
*Yummy prépare les meilleurs plats pour vos fêtes*  
*****
```

Et enfin, nous devons afficher la nouvelle devise entouré de Ss

```
public class Yummy2 {  
    Run | Debug  
    public static void main(String[] args) {  
        System.out.println("SsSsSsSsSsSsSsSsSsSsSsSsSsSsS");  
        System.out.println("sYummy c'est fun au soleils");  
        System.out.println("SsSsSsSsSsSsSsSsSsSsSsSsSsSsS");  
    }  
}
```

```
SsSsSsSsSsSsSsSsSsSsSsSsSsSsS  
sYummy c'est fun au soleils  
SsSsSsSsSsSsSsSsSsSsSsSsSsSsS
```

Conclusion :

Ce TP nous a permis de voir comment fonctionne le java et de mettre nos algorithmes en code. Pour ma part, j'ai trouvé ce tp plus compliqué car je n'avais jamais réellement utilisé de java. La partie la plus dure a sûrement été le debug 4 ou il a fallu se renseigner sur le void et la création du "jeu" qui m'a pris du temps car je ne savais pas du tout quels balise/commande utilisés.